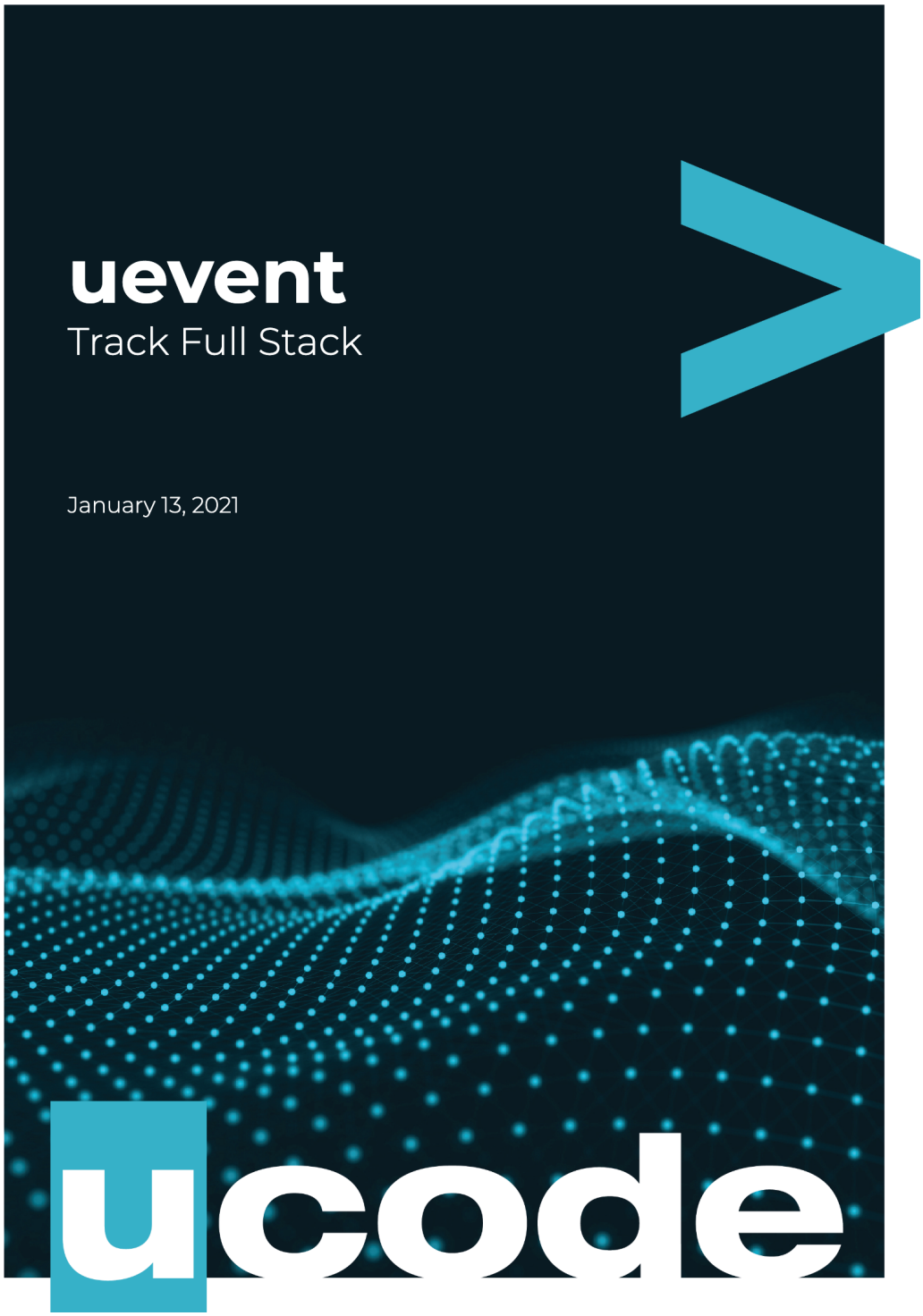
uevent
Track Full Stack
January 13, 2021
ucode

Contents

Engage

DESCRIPTION

Hello, everyone!

Imagine the situation. You want to attend an event in another city. You buy tickets for the event, find the best way to travel, search for accommodation. After all this, you are alone all day or half-day before the beginning of the event for which you are there. Once the event is finished, you notice some of your acquaintances. But you could travel together and spend all this time together or even split the expenses for accommodation!

People need a way to search for an interesting event, whether it be a scientific conference or a stand-up show, and get to know their friend is going to visit it too or find new people before the event.

Finding people with the same interests isn't always that easy. Finding people with the same goals and aspirations is even harder. It's not about meeting just one specific person who shares all your interests, but about opening yourself to different people who share at least one common interest. Meeting like-minded people isn't hard - it's about taking the right steps.

This challenge combines the improvement of development skills that you received from previous challenges and gaining new skills, both technical and business-related. Here are some of them:

- implementation of a third-party API
- work with the map
- CSS skills
- implementation of a payment system
- notifications
- implementation of a shopping cart
- UI/UX skills
- time management
- product thinking
- much more you and the team decide to achieve

This challenge is creative enough to bring all your wildest ideas to life!

BIG IDEA

Connecting like-minded people.

ESSENTIAL QUESTIONS

How to bring together people with similar concerns?

CHALLENGE

Create a service to unite people with the same interests.

uevent | Track Full Stack **> 2**

ucode

Investigate

GUIDING QUESTIONS

We invite you to find answers to the following questions. By researching and answering them, you will gain the knowledge necessary to complete the challenge. To find answers, ask the students around you and search the internet. We encourage you to ask as many questions as possible. Note down your findings and discuss them with your peers.

- What are the main principles of UI/UX?
- What questions do you need to ask yourself to make a great UI?
- What is `REST` concept?
- What is a `RESTful` web service?
- What is SOAP? What is the difference between `SOAP` and `REST`?
- Why do you need to write documentation about your service work?
- Why must your service support both formats: `JSON` and `XML`?
- What is a git-flow?
- What is an `SQL injection`? How to defend your service from this attack?

GUIDING ACTIVITIES

Complete the following activities. Don't forget that you have a limited time to overcome the challenge. Use it wisely. Distribute tasks correctly.

- Meet the team.
- Read the story.
- Investigate which tasks take the most time during development.
- Investigate why planning may help to save time and make error-free web services.
- Implement advantages of planning in your teamwork.
- Investigate how to keep the user's attention on the site.
- Explore whether unit tests are useful and how they save time in the development process.
- Try to choose the best solutions.
- Distribute tasks between all team members.
- Clone your git repository that is issued on the challenge page.
- Start to develop the solution. Offer improvements. Test your code.
- Explore new things. Talk, discuss, and communicate.

ucode

uevent | Track Full Stack **> 3**

ANALYSIS

Analyze your findings. What conclusions have you made after completing guiding questions and activities? In addition to your thoughts and conclusions, here are some more analysis results.

- Challenge has to be carried out by the entire team.
- Each team member must understand the challenge and realization, and be able to reproduce it individually.
- It is your responsibility to assemble the whole team. Phone calls, SMS, messengers are good ways to stay in touch.
- Be attentive to all statements of the story.
- Analyze all information you have collected during the preparation stages. Try to define the order of your actions.
- You can proceed to `Act: Creative` only after you have completed all requirements in `Act: Basic`. But before you begin to complete the challenge, pay attention to the program's architecture. Take into account the fact that many features indicated in the `Act: Creative` require special architecture. And in order not to rewrite all the code somewhen, we recommend you initially determine what exactly you will do in the future. Note that the `Act: Basic` part gives the minimum points to validate the challenge.
- Validate all input fields with relevant error messages.
- Check whether all possible errors are properly handled.
- Your service must not crash in any way.
- Make sure that your web app is responsive and is correctly displayed on different screen resolutions (mobile, tablet, etc.).
- Complete tasks according to the rules specified in the following style guides:
  - HTML and CSS: Google HTML/CSS Style Guide. As per section 3.1.7 Optional Tags, it doesn't apply. Do not omit optional tags, such as `<head>` or `<body>`
  - JavaScript:
    * JavaScript Style Guide and Coding Conventions
    * JavaScript Best Practices
    * Airbnb React/JSX Style Guide
    * Node.js Best Practices
- The solution will be checked and graded by students like you. Peer-to-Peer learning.
- If you have any questions or don't understand something, ask other students or just Google it.

uevent | Track Full Stack **> 4**

Act: Basic

ALLOWED

HTML, CSS, JS (Node, React), database of your choice

DESCRIPTION

The ability to communicate with other people is one of the major skills in the modern world. To improve this skill, you must constantly interact with different people from diverse spheres. That's why this challenge is socially useful.

Create a service that helps people broaden their horizons and improve their networking skills. Find services that give information about entertaining/educational events in any chosen city or country, and use them as an example.

But remember that creativity is the key challenge for every solution you work on, so do not copy somebody else's interface design or logic.

As usual, your service must be available for everybody. There should be different access levels and different features accessible for registered and non-registered users.
Your service must provide a possibility to see upcoming events.

The main page will be the list of events. It could be not sorted at all or sorted by the format or by the theme of the event. Possible formats include conferences, lectures, workshops, fests, etc. And themes - business, politics, psychology, etc. Users must have the possibility to choose the category of viewed events. It is also very convenient to sort events by date.

Every event page must possess such `features`:

- subscription to the event
- subscription to notifications from the organizer of the current event

Every event page must contain the following `elements`:

- description of the event
- price of the ticket (it can be free, or the user can apply a promo code)
- list of users who will come to the event
- comments for the event from other users or organizers
- information about the organizer
- other events of the organizer
- similar events

Also, you must show the location of the event on the map.

Integrate the payment system that allows the user to pay by card without leaving your service. Research ways to implement a payment system. You will find out that it can be done from the localhost IP. That's why you need to investigate which response you could possibly get from the payment service API and just fake it in case the user wants to pay

uevent | Track Full Stack **> 5**

on your service by card. In the future, when you decide to present your service to the general public, you will receive real API responses that would be already handled.
Once the user pays for the ticket, the email notification with automatically generated tickets must be sent. Service has to send a reminder when the time of the user's event is coming.

For the user account you must implement such features :

- admin panel
- user's events
- user's tickets
- user's notifications
- user's profile setting (think about which fields could be editable)

The user should be able to choose whether to show their name on the list of event visitors or not. The design of the user profile page is fully up to you. Only signed up users can buy tickets and register to events, but you can choose which other features or pages are restricted for non-registered users. You can show them only the list of events or you can let them look through details of a certain event.

COMPANIES

Users must have the possibility to register a company on your web service. It means that user will be able to create events on behalf of this company and add news for their company.

Only users who created the company entity on their profiles will have the possibility to create an event.

Think over which data can be needed for the company entity. It would be logical to include at least the company name, email and location.

It is clear that the amount of tickets for the event could be limited. That's why you must pay attention to this while implementing the event creation page. Think over which data can be needed on the event creation page. Also on this page, the user with the created company must have the possibility to:

- select and upload a poster for an event (if it wasn't set, there must be a default one)
- choose whether the user wants to receive notifications about new visitors of their own event
- choose who can see the list of event visitors (everybody or only users who are going to the event)
- choose the date when the user's event will be published on the service
- create a promo code (or a set of codes) that the users could apply in the future to cut the ticket price
- set the page where the potential visitor will be redirected after they buy a ticket

uevent | Track Full Stack **> 6**


It is useful and considered a good tone to add small hints for the users near the fields
to avoid misunderstandings.
ucode
uevent | Track Full Stack > 7

Act: Creative

DESCRIPTION

It is the place where your imagination and creativity plays a major role. Implement additional features to make the program better and more unique. Listed below are a few ideas you can add to your program. You can come up with everything you want to improve your program. Creative features:

- a separate page for creating notifications:
  - there would be a possibility to choose for whom you want to send a reminder
  - choose the way how to send the reminder (by email, by web push or via telephone)
  - choose a theme of the reminder and the reminder body
- possibility to log in for a user via other social services (Gmail, Facebook, LinkedIn, etc.)
- change the language of displaying your service (to English, for example, or any other language)
- integrate a unique QR-code in every ticket
- integrate your `chronos` so that events to which you have subscribed automatically appear in your default calendar
- other creative features

uevent | Track Full Stack **> 8**

Document

DOCUMENTATION

One of the attributes of Challenge Based Learning is documentation of the learning experience from challenge to solution. Throughout the challenge, you document your work using text and images, and reflect on the process. These artifacts are useful for ongoing reflection, informative assessment, evidence of learning, portfolios, and telling the story of challenge. The end of each phase (Engage, Investigate, Act) of the challenge offers an opportunity to document the process.

Much of the deepest learning takes place by considering the process, thinking about one's own learning, analyzing ongoing relationships with the content and between concepts, interacting with other people, and developing a solution. During learning, documentation of all processes will help you analyze your work, approaches, thoughts, implementation options, code, etc. In the future, this will help you understand your mistakes, improve your work, and read the code.

At the learning stage, it is important to understand and do this, as this is one of the skills that you will need in your future job. Naturally, the documentation should not be voluminous, it should be drawn up in an accessible, logical, and connected form.

So, what must be done?

- a nice-looking and helpful README file. In order for people to want to use your product, their first introduction must be through the README on the project's git page. Your README file must contain:
  - Short description. This means, that there must be some info about what your project actually is. For example, what your program does.
  - Screenshots of your solution. This point is about screenshots of your project "in use".
  - Requirements and dependencies. List of any stuff that must be installed on any machine to build your project.
  - How to run your solution. Describe the steps from cloning your repository to the first launch of your program.
- a full-fledged documentation in any forms convenient for you. By writing this, you will get some benefits:
  - you have an opportunity to think through implementation without the overhead of changing code every time you change your mind about how something should be organized. You will have very good documentation available for you to know what you need to implement
  - if you work with a development team and they have access to this information before you complete the project, they can confidently start working on another part of projects that will interact with your code
  - everyone can find how your project works
- your documentation must contain:
  - Description of progress after every competed CBL stage.
  - Description of the algotithm of your whole program.

uevent | Track Full Stack **> 9**


Keep in mind that the implementation of this stage will be checked by peers at the assessment!
Also, there are several links that can help you:
• Make a README
• How to write a readme.md file?
• A Beginners Guide to writing a README
• Google Tools - a good way to journal your phases and processes:
– Google Docs
– Google Sheets
• Dropbox Paper - a tool for internally developing and creating documentation
• Git Wiki - a section for hosting documentation on Git-repository
• Haroopad - a markdown enabled document processor for creating web-friendly documents
• Canva - a good way to visualize your data
• QuickTime - an easy way to capture your screen, record video or audio (macOS)
• ScreenToGif - screen, webcam, and sketchboard recorder with an integrated editor (Windows)
• code commenting - source code clarification method. The syntax of comments is determined by the programming language
• and others to your taste
ucode
uevent | Track Full Stack > 10

Share

PUBLISHING

Last but not least, the final stage of your work is to publish it. This allows you to share your challenges, solutions, and reflections with local and global audiences. During this stage, you will discover ways of getting external evaluation and feedback on your work. As a result, you will get the most out of the challenge, and get a better understanding of both your achievements and missteps.

To share your work, you can create:

- a text post, as a summary of your reflection
- charts, infographics or other ways to visualize your information
- a video, either of your work, or a reflection video
- an audio podcast. Record a story about your experience
- a photo report with a small post

Helpful tools:

- Canva - a good way to visualize your data
- QuickTime - an easy way to capture your screen, record video or audio (macOS)
- ScreenToGif - screen, webcam, and sketchboard recorder with an integrated editor (Windows)

Examples of ways to share your experience:

- Facebook - create and share a post that will inspire your friends
- YouTube - upload an exciting video
- GitHub - share and describe your solution
- Telegraph - create a post that you can easily share on Telegram
- Instagram - share photos and stories from ucode. Don't forget to tag us :)

Share what you've learned and accomplished with your local community and the world. Use #ucode and #CBLWorld on social media.

ucode

uevent | Track Full Stack **> 11**